

## Remarks

The Office Action mailed on April 4, 2003 has been fully considered.

The Office Action rejects claims 1-29 under 35 U.S.C. § 103 as being obvious over Van Huben et al. (U.S. Pat. No. 6,484,177). Van Huben discloses a method of enabling a directory service to interact with a central data management system (DMS) by employing a single access application at a user API to access a heterogeneous data storage system comprising data objects that are classified according to a PFVL paradigm (Package, Filetype, Variance, and Level). In particular, the Van Huben invention is entirely based on imposing a consistent PVFL view on the heterogeneous data storage systems such as LDAP, relational databases, and PDM. (Col. 5, lines 8-10; Col. 6, lines 34-37, 63-66; Col. 7, lines 34-39, 66-67; and Col. 7, lines 1-33).

The present invention, on the other hand, discloses a system for providing a single, uniform, and consistent API for accessing and managing content in multiple, disparate repositories that does not employ a particular type of data classification such as PVFL. Rather than adding the access, exchange, and management capabilities that are typically found in document management systems to the objects residing in directory services as taught by Van Huben, the present invention exposes those capabilities as they *already exist* in the document management systems. In fact, the invention as taught by Van Huben would not support document management systems that already include those capabilities and therefore could not be substituted for the present invention.

Referring now to the rejection of independent claims 1, 14, and 22, the Office Action asserts that the client/server interface of Van Huben is a view services component for reviewing the results generated from a user request. The Office Action admits that Van Huben does not teach converting and processing the individual items from the query results

into a format understandable by the API and asserts it would have been obvious for the Van Huben command translators to perform this function since they convert the user request into a format understandable by the repositories.

First, Applicant respectfully submits that Van Huben does not teach or suggest converting and processing the individual items from the query results into a format understandable by the API. The mere disclosure that the Van Huben command translators translate a user request into a format understandable by the disparate repositories does not suggest that command translators also convert and process individual items from the query results into a format understandable by the requesting client API.

Second, the view services component of the present invention converts and processes query results into a format that is supported by the *viewing technologies* of the particular client application that generated the request. Thus, the view services component may convert and process the results into different formats depending on what is supported by the client API. *See* Application, page 9, lines 17-20 (“Therefore, view services 106 is able to display content from the repositories 110 without any repository-specific client software or image viewer being installed on the user’s computer.”); *see also* Application, pages 13-15 (discussing conversion and processing of content from a repository into a format based on the content’s mime type). This feature is not taught or suggested by Van Huben.

Independent claims 1, 14, and 22 have been amended to more clearly recite that the view services component processes and converts query results into a format that is supported by the viewing capabilities of the client API. The independent claims, as amended, are patentable over Van Huben. Dependent claims 2-13, 17-21, and 23-29 are patentable for the same reasons regarding the independent claims.

With regard to claims 4, 17, and 25, Van Huben discloses in Col. 14, lines 56-59 and Fig. 4B how data is exposed in the PVFL paradigm from the data repositories and how the data may be in an XML format. This disclosure does not teach or suggest the import and export of non-XML content and metadata properties in an XML format, as taught by the present invention. (*See* Application, page 9, lines 22-24 (“The exchange services component 116 enables interaction between enterprises by providing seamless export and import of content in disparate repositories 110, along with associated metadata properties, in an XML format.”)) Claims 4, 17, and 25, as amended, more clearly recite this aspect of the invention and are therefore patentable over Van Huben.

Regarding claim 5, Van Huben does not teach or suggest that the client API is in a Java, COM, or web service format. The Office Action’s references to disclosure of these formats in Van Huben is misplaced because those disclosures relate to the multiple format types for the data in the repositories, not the types of formats of the client API.

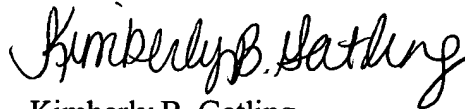
As discussed above, Van Huben does not teach or suggest a view services component as claimed in the present application. Accordingly, Van Huben does not teach or suggest a view services component as an Enterprise Java Bean (EJB), as recited in claim 7. Also, Van Huben does not teach or suggest that each bridge (command translator) is an EJB, as recited in claim 8. In fact, there is no reference in Van Huben to an EJB. Therefore, claims 8 and 9 are patentable over Van Huben.

Regarding claims 9, 19, and 27, Van Huben discloses command translators that translate user requests into a format understandable by the disparate repositories but Van Huben does not teach or suggest a factory for creating new translators when needed, as recited in the claims of the present invention. Therefore, claims 9, 19, and 27 are patentable over Van Huben.

Regarding claim 13, as noted above, Van Huben discloses how data is exposed in the PVFL paradigm from the data repositories and how the data may be in an XML, Java, JPG, or HTML format. This disclosure, however, does not teach or suggest that each bridge (command translator) *accesses* its underlying content repository via Java, Component Object Model (COM), or Java Native Interface (JNI) application program interface (API), as recited in the claim. Further, the DMS capabilities disclosed in Col. 11, lines 5-7 of Van Huben are not directed towards the format used to *access* the underlying content in the repositories. Therefore, claim 13 is patentable over Van Huben.

Based on the foregoing amendments and remarks, allowance of the claims in the application is respectfully requested.

Respectfully submitted,



Kimberly B. Gatling  
Reg. No. 48,814

SMITH MOORE LLP  
P.O. Box 21927  
Greensboro, NC 27420  
(336) 378-5356

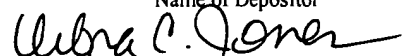
Date: September 4, 2003  
File No.: 5003397-100US1

CERTIFICATE OF MAILING

I HEREBY CERTIFY THAT THIS  
DOCUMENT IS BEING DEPOSITED WITH  
THE UNITED STATES POSTAL SERVICE AS  
EXPRESS MAIL, IN AN ENVELOPE  
ADDRESSED TO: Mail Stop Non-Fee  
Amendment; COMMISSIONER FOR  
PATENTS, P.O. Box 1450, Alexandria, 22313-  
1450, ON \_\_\_\_\_  
(Date of Deposit)

Debra C. Jones

\_\_\_\_\_  
Name of Depositor



\_\_\_\_\_  
Signature

9/4/03

\_\_\_\_\_  
Date of Signature